2024 European Control Conference (ECC)
June 25-28, 2024. Stockholm, Sweden

Collision Avoidance for ASVs in Archipelagos—A COLREGs-Aware Optimization-Based Method

Birgitta Wingqvist[1,2], Björn Olofsson[1], Jens-Olof Lindh[2], Anders Robertsson[1], and Rolf Johansson[1]

Abstract— **With increased autonomy in marine vessels, autonomous surface vessels (ASVs) and conventionally manned vessels need to coexist at sea. Any relocation needs to include collision avoidance according to the traffic rules at sea, COLREGs. Here, a local collision-avoidance planner for an archipelago environment is presented. The optimization-based local planner presented considers a predefined nominal path and upon detection of other vessels, a COLREGs-aware maneuver is computed. The maneuver adapts to the available space and includes a return to the nominal plan. The irregular available space for maneuvers is approximated with a local convex area. The planner was evaluated both in simulations and in field experiments.**

I. INTRODUCTION

The introduction of autonomy in vessels at sea is motivated by a potential to increase safety and efficiency. Automatic functions may, for example, be introduced as support functions for the crew or as tasks performed fully autonomously by the vessel. In any task or mission at sea that includes relocating the vessel, the motion planner would need to adhere to the International Regulations for Preventing Collisions at Sea COLREGs [1], [2].

From a motion planning and control perspective, previous research mainly considered COLREGs with respect to Rule 7–8, 13–17 [3]. These are also the rules considered here and describe the need for determining risk of collision (Rule 7), actions to avoid collision (Rule 8), as well as the applicable evasive maneuvers for overtake (Rule 13), head-on (Rule 14) and crossing situations regarding power-driven vessels (Rules 15–17) under good visibility conditions. COLREGs do not provide an absolute answer or precise distances, which means that the rules need to be interpreted and evaluated before implementation in an automatic function. This stage setting also poses uncertainty in how to evaluate an action in terms of safety and efficiency. Research on algorithm metrics and evaluation can be found in, *e.g.*, Stankiewicz *et al.* [4] and Vagale *et al.* [5].

It should be noted that COLREGs compliance is not solved solely by implementing an algorithm taking the COLREGs rule set mentioned in the previous paragraph into account. Reliable situation awareness is a prerequisite, but also more general challenges exist. Weber *et al.* noted that the current algorithms may pose worse predictability, compared to only conventionally manned vessels, as the tuning of parameters may significantly impact the behavior [6]. Zhou *et al.* highlighted potential challenges that are introduced with the concept and definition of visibility states [7]. In [8], MacKinnon *et al.* discussed the existence of multiple possible actions, especially when the planned route is not a straight line, where the possible use of voice over radio would encourage vessel collaboration.

In [3], articles on collision avoidance and path planning techniques at sea are evaluated and surveyed, with main focus on COLREGs coverage and limitations, but also on applicability in different environments. One of the more common planning approaches is to use predictive control, *i.e.*, to predict the outcome of various actions and choose the action with the best prediction score. In the survey [3], it was concluded that the majority of articles considered open water, even though coastal areas also are considered to a large extent. However, the denser the environment, the lower the amount of articles seems to be. Sweden, where this research was performed, has plentiful of archipelago that defines the marine environment. As opposed to open water, the archipelago limits the maneuver space.

Previous research on predictive control includes the work by Hagen *et al.* with a discrete predictive controller designed for open water [9]. Stankiewicz *et al.* proposed a predictive planner, demonstrated in an open water setting, utilizing a cost function for obtaining *good seamanship* [10]. For use in more restricted water, Eriksen *et al.* proposed a layered predictive planner and controller [11]. They presented a path-following collision-avoidance system that considers stand-on situations, readily apparent actions, and the presence of static obstacles such as simulated elliptic islands. In [12], Bergman *et al.* proposed a two-level planner for a complex environment, such as an archipelago, where the first level is a lattice-based planner that evaluates long-term considerations of the plan, including collision avoidance. The second level is an optimal controller that further evaluates and executes the long-term plan using a receding horizon.

II. PROBLEM STATEMENT

The problem considered here is autonomous transition and motion planning for collision avoidance in an archipelago with islands, shallows, other static obstacles, as well as dynamic obstacles such as other (power-driven) vessels. The aim is to investigate and suggest solutions to some of the challenges the environment implies on a predictive planner, and more specifically:

1) How to model and integrate the irregularly shaped environment into a problem that can be efficiently solved by a numerical solver?

[1]Department of Automatic Control, Lund University, PO Box 118, SE-221 00 Lund, Sweden. The authors are members of the ELLIIT Strategic Research Area at Lund University. (birgitta.wingqvist@control.lth.se)
[2]Saab Kockums AB, SE-205 55 Malmö, Sweden.

2) How to adapt the problem to the varying space?
3) How to incorporate COLREGs into the problem?
4) How can a non-straight nominal path be incorporated and accounted for with regards to the stand-on situation, predictability, and mission efficiency?

The ego vessel is assumed to have a predefined mission, here defined in the form of an initial nominal path and a reference speed. The nominal path defines where to go and between which islands. When no other vessels are around, the ego vessel may focus on following the nominal path, but once a nearby vessel is detected, the priority is shifted toward behaving predictable and avoiding collision. Predictability here means to make readily apparent actions, if any, and to perform the evasive maneuvers as per COLREGs. The vessels are assumed to be smaller boats, such that boat dynamics and size can be negligible in relation to the larger surroundings. The static environment is assumed known and all applicable vessels are assumed to be identified with current position, course, and speed, as long as within line of sight from the ego vessel, *i.e.*, identification of other vessels, *e.g.*, using AIS is not to be expected.

III. Method

A local planner is introduced that solves an optimization problem in order to find the best maneuver. The planner should aim to move the vessel forward along the nominal path, but prioritize actions needed to avoid potential collisions. Actions here both mean active avoidance and the ability to keep course and speed when the situation so requires. A system overview is presented in Fig. 1.

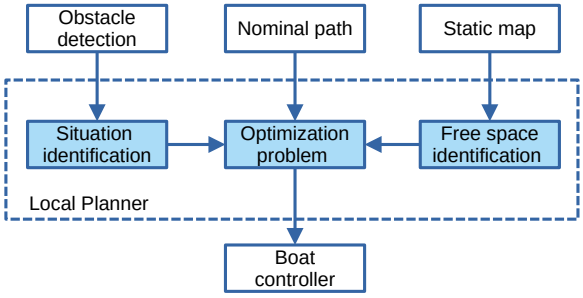

Fig. 1. Overview of the components and interfaces of the local planner.

The core of the local planner is an optimization problem designed for finding the optimal inputs, relying on model prediction, further described in Sec. III-A. Two categories of space restrictions are identified. The first category is large static irregular areas such as islands and shallows. These are accounted for by creating a local convex free space $\mathbf{X}_\mathrm{f}$ as input to the planner (Sec. III-B). The second category is smaller static or dynamic obstacles such as markers or other vessels. These are mathematically modeled as elliptic areas that have a cost associated with intrusion (Sec. III-D). For obstacle vessels, the shape of the restricted area depends on the applicable COLREGs situation (Sec. III-C).

The resulting plan consists of timed way-points in coordinates of north, N_k, and east, E_k, with action cycle T_a. The start would be in the current position $\mathbf{x}_\mathrm{s}$ and the end at a pursuit point (far) ahead on the nominal path, $\mathbf{x}_\mathrm{pp}$. The timed way-points are linked together with a corresponding course over ground, $u_{\psi,k}$, and speed over ground, $u_{v,k}$, such that

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k, T_\mathrm{a}) \tag{1}$$

where $\mathbf{u}_k = (u_{\psi,k}, u_{v,k})^T$, $\mathbf{x}_k = (N_k, E_k)^T$, and

$$f(\mathbf{x}_k, \mathbf{u}_k, T_\mathrm{a}) = \mathbf{x}_k + T_\mathrm{a} u_{v,k} \begin{pmatrix} \cos u_{\psi,k} \\ \sin u_{\psi,k} \end{pmatrix} \tag{2}$$

The varying environment in an archipelago impacts the planning horizon in time and distance. In Fig. 2, the action instants correspond to the dots. They need to be close enough not to let the elliptic obstacle pass by unnoticed in between any of the dots. On the other hand, the planning horizon itself needs to be long enough to take the full action, and possible interactions, into account. By introducing T_a as a variable and not a constant, the horizon for the plan in time, and distance, is variable (see, *e.g.*, [13] where Rösmann *et al.* presented a timed-elastic band). The number of prediction steps H_p is selected large enough to cover the horizon well and small enough to keep the computational time practical. With constant speed, the evaluation points will be evenly spread out in time.

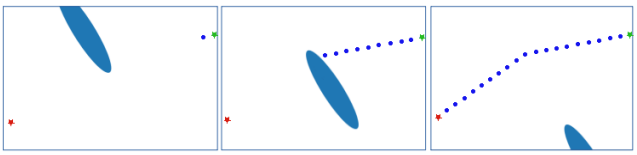

Fig. 2. Illustration of an evasive trajectory plan at three different times over the prediction horizon. The current position is the green dot at the upper right and the goal position, *i.e.*, the pursuit point, is the red dot in the lower left. The obstacle is associated with a moving elliptic restricted area.

A. The Optimization Problem

The optimization problem proposed to be solved at a planning interval T_I over a horizon H_p, given a reference speed v_r, is formulated as

$$\begin{aligned}
&\underset{\mathbf{u}_1, \ldots, \mathbf{u}_{H_\mathrm{p}}}{\text{minimize}} \quad T_\mathrm{a} + J_v + J_\mathrm{obs} \\
&\text{subject to} \\
&\quad \mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k, T_\mathrm{a}) && k = 1, \ldots, H_\mathrm{p} \\
&\quad h_{n,k}(\mathbf{x}) \leq 0 && n = 1, \ldots, N, \\
&\qquad && k = 2, \ldots, H_\mathrm{p}+1 \\
&\quad u_{v,k} \leq v_\mathrm{r} + \sigma_v && k = 1, \ldots, H_\mathrm{p} \\
&\quad \mathbf{x}_k \in \mathbf{X}_\mathrm{f} && k = 2, \ldots, H_\mathrm{p} \\
&\quad T_\mathrm{a} H_\mathrm{p} \geq T_\mathrm{min} \\
&\quad \sigma_v \geq 0, \quad 0 \leq \sigma_\mathrm{obs} \leq 1 \\
&\quad \mathbf{x}_1 = \mathbf{x}_\mathrm{s}, \quad \mathbf{x}_{H_\mathrm{p}+1} = \mathbf{x}_\mathrm{pp}
\end{aligned} \tag{3}$$

where the involved variables are defined and explained in the previous and following sections. The cost function consists of three parts with the action cycle T_a being the first. By

minimizing T_{a}, the total time to reach the pursuit point $\mathbf{x}_{\mathrm{pp}}$, given the constraints, is minimized. By selecting $T_{\min} > T_{\mathrm{I}}$ it is also ensured that the plan will last at least until the next planning iteration. If space is limited, the speed will decrease accordingly. Secondly, the penalty function for speed, J_v, considers both a speed increase above v_{r}, with the introduction of a slack variable σ_v, and a smaller penalty for speed itself. The latter term helps to keep the plan well-stretched in space. The expression is

$$J_v = \gamma_{v1}\sigma_v^2 + \gamma_{v2}\sum_{k=1}^{H_{\mathrm{p}}} u_{v,k}^2 \tag{4}$$

where γ_{v1} and γ_{v2} are weights selected such that $\gamma_{v2} \ll \gamma_{v1}$. An increase in speed should not be allowed in order to reach the defined pursuit point faster, but only if needed in order to ensure a collision-free path forward. The weight γ_{v2} is therefore proportionally large. Thirdly, obstacles shall be avoided. Elliptic restricted areas were introduced in [11] and the elliptic constraint equation as such is adopted here. The modification here of the constraint is the introduction of a slack variable σ_{obs} as

$$h_{n,k}(\mathbf{x}) = -\ln\left[\frac{(\tilde{N}_{n,k}^c\cos\theta_n + \tilde{E}_{n,k}^c\sin\theta_n)^2}{(r_{x,n}(1-\sigma_{\mathrm{obs}}))^2} + \frac{(\tilde{N}_{n,k}^c\sin\theta_n - \tilde{E}_{n,k}^c\cos\theta_n)^2}{(r_{y,n}(1-\sigma_{\mathrm{obs}}))^2} + \varepsilon\right] + \ln(1+\varepsilon) \tag{5}$$

where ε is a small numerical constant used to avoid singularities. $\tilde{N}_{n,k}^c$ and $\tilde{E}_{n,k}^c$ define the positional difference between the planned position of the ego vessel and the center of the ellipse associated with obstacle n at time k as per

$$\tilde{N}_{n,k}^c = N_k - N_{n,k}^c, \quad \tilde{E}_{n,k}^c = E_k - E_{n,k}^c \tag{6}$$

where $N_{n,k}^c$ and $E_{n,k}^c$ define the center of the ellipse. Note that the center of the ellipse may differ from the obstacle position, as further described in Sec. III-D. The course of obstacle n is defined by θ_n, the radius along the direction of travel is $r_{x,n}$, and $r_{y,n}$ is the radius along the perpendicular direction. The predicted movement of an obstacle is based on its current speed and course, and in the case that the obstacle is static, $v_n = 0$. The movement of the ellipses over time is computed as

$$N_{n,k}^c = N_{n,0}^c + kT_{\mathrm{a}}v_n\cos\theta_n \tag{7a}$$

$$E_{n,k}^c = E_{n,0}^c + kT_{\mathrm{a}}v_n\sin\theta_n \tag{7b}$$

The slack variable $\sigma_{\mathrm{obs}} \in [0,1]$ is common to all obstacles, and by minimizing this variable the maximum intrusion into any of the restricted areas is minimized. Again, as seen in Fig. 5, the obstacle itself may not be in the center of the restricted area and therefore the cost should be high to avoid any intrusion. Using γ_{obs} as a weight, the cost is defined by

$$J_{\mathrm{obs}} = \gamma_{\mathrm{obs}}\sigma_{\mathrm{obs}} \tag{8}$$

B. Identifying the Available Free Space and Pursuit Point

By using a static map with irregularly shaped areas, marking free and non-free space, the free space is converted to optimization-friendly local convex inner approximations. Such an approximation is motivated by that the surroundings, including the presence of other vessels, are only considered known within the local visual area and planning beyond is not applicable. The approximated area of free space is introduced as half-plane constraints in the problem as

$$\mathbf{X}_{\mathrm{f}} = \{\mathbf{x} \in \mathbb{R}^2 | \mathbf{A}\mathbf{x} \leq \mathbf{b}\} \tag{9}$$

The creation of such a space has previously been performed by expanding a boat-like shape in [12], [14], or by using tangents to detected obstacles as in [15]. Both of these expansion schemes are ego-vessel centered. Here, $\mathbf{X}_{\mathrm{f}}$ is instead created by an heuristic expansion scheme focused around the nominal path. With $\mathbf{q}_{\mathrm{p}}$ defining the north and east coordinates of the path at a path coordinate s_{p}, the expansion scheme is summarized as follows and visualized in Fig. 3. First, define the relevant path points:

1.1 Identify the current path point $\mathbf{q}_0$ and path parameter s_0 along the path.

1.2 Identify how far along the path the visibility reaches, s_{vis}, and find $s_{\max}$ by $s_{\max} = \max(s_0 + l_{\max}, s_{\mathrm{vis}})$. The parameter $l_{\max}$ defines the maximum look-ahead distance.

1.3 Identify the nominal path expansion point, $\mathbf{q}_{\mathrm{exp}}$, by $s_{\mathrm{exp}} = (s_0 + s_{\max})/2$.

Second, step-wise expand the free space. In each step, the expansion is checked for convexity and intersections with the static map. The expansion is done by adding points and checking if the resulting convex hull is intersecting with any of the polygons describing the restricted areas:

2.1 Add a smaller ego-centered space to include the ego vessel forward direction of movement.

2.2 Expand space, along the nominal path, from s_0 up to s_{exp}. If the vessel has a stand-on obligation, simultaneously add points along the stand-on path (see Sec. III-E). If either direction is not possible to expand further, then stop the expansion and set the last point included in this step on the stand-on path to be $\mathbf{q}_{\mathrm{SO,exp}}$ as in Fig. 6.

2.3 Step-wise expand along five directions. The first four directions are the expansion of a box centered at $\mathbf{q}_{\mathrm{exp}}$ (or $\mathbf{q}_{\mathrm{SO,exp}}$), with an expansion limit of $l_{\max}$, and the fifth direction is along the nominal path, toward $\mathbf{q}_{\max}$. For each expansion $j \in [1,5]$, expand a step d_j along the corresponding direction, if it is not possible decrease the step size d_j in the next iteration. Stop expansion j if d_j is smaller than a defined value, here 5 m was used.

To avoid numerical issues in the path inclusion, add a corresponding corridor around the path. It also proved useful to the numerical solver to move the pursuit point, red in Fig. 3, slightly into the free space, away from the edge but still along the path.

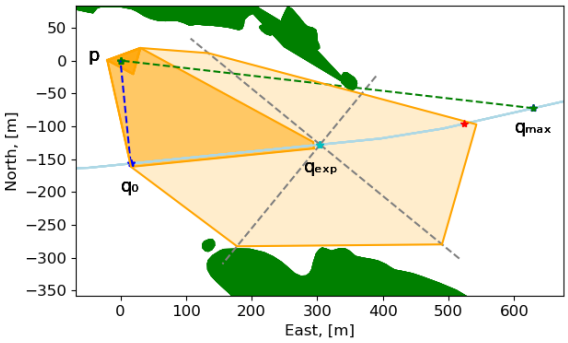

Fig. 3. The available free space is approximated with a path-centered convex space. The different shades of orange illustrate the different expansion steps. The resulting pursuit point is marked in red.

C. Situation Identification

For obstacle vessels within the visible range, the current state (including position, course, and speed over ground) is considered system input. In order to identify the applicable COLREGs situation (if any) for an obstacle vessel, the relation to the current state of the ego vessel is evaluated. The situations considered are overtake (OT), head-on (HO), give-way when crossing (GW), stand-on when crossing (SO), as well as an emergency situation (EM). If none of these situations is applicable the situation is considered safe (SF). The situation identification is done by first evaluating the collision risk based on a closest-point approach, as also done in [11] and [12]. Given the assumption of constant course and speed for all involved vessels, the time to reach the closest point of approach, t_{CPA}, is for obstacle n computed by using projection of the relative speed vector $\tilde{\mathbf{v}}_n$ on the relative position vector $\tilde{\mathbf{p}}_n$. The closest expected distance, d_{CPA}, between the vessels is the distance between the vessels at t_{CPA} according to

$$t_{\mathrm{CPA},n} = \frac{\tilde{\mathbf{p}}_n \cdot \tilde{\mathbf{v}}_n}{|\tilde{\mathbf{v}}_n| + \varepsilon}, \quad d_{\mathrm{CPA},n} = |(\tilde{\mathbf{p}}_n + t_{\mathrm{CPA},n}\tilde{\mathbf{v}}_n)| \tag{10}$$

To prevent division by zero, a small value ε is introduced. If $t_{\mathrm{CPA}} < 0$, this means that the vessels are not approaching.

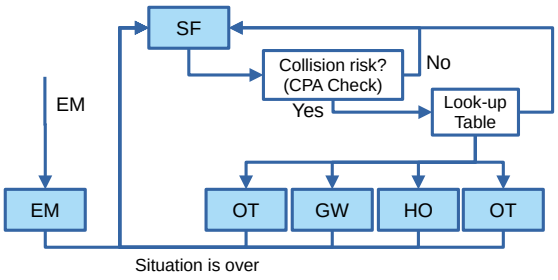

Fig. 4. Situation identification. For a non-safe state, the situation needs to be resolved before it can reach another state. The detection of emergency state, EM, will override any other identification.

The situation identification for each obstacle vessel is a state machine as illustrated in Fig. 4 and it is run continuously in between the planning instants. First, the collision risk is evaluated. The obstacle vessel is considered to be at a safe distance if $(d_{\mathrm{CPA}} > d_{\mathrm{SF}}) \vee (t_{\mathrm{CPA}} > t_{\mathrm{SF}})$, where d_{SF} and t_{SF} are parameters. Should the combined t_{CPA} and d_{CPA} trigger a potential risk, the situation is further evaluated according to a look-up table. The table uses relative bearing and relative course as look-up parameters. Relative bearing for vessel B to vessel A is defined as

$$\phi_{\mathrm{AB}} = \mathrm{atan2}(E_{\mathrm{B}} - E_{\mathrm{A}}, N_{\mathrm{B}} - N_{\mathrm{A}}) - \theta_{\mathrm{A}} \tag{11}$$

with $\phi_{\mathrm{AB}} \in [0, 2\pi[$ and correspondingly for ϕ_{BA}. The relative course is defined as

$$\theta_{\mathrm{AB}} = \theta_{\mathrm{B}} - \theta_{\mathrm{A}}, \quad \theta_{\mathrm{AB}} \in [0, 2\pi[\tag{12}$$

The look-up table, not further defined here, extracts the situation and is based on [16]. However, the selection scheme is here simplified as all vessels located aft of the ego vessel are here defined as SF, and OT can only occur if the speed of the ego vessel is higher than the speed of the obstacle vessel. In addition, the angle for θ_{AB} defining the OT situation has been decreased, compared to [16], from $\pm 3\pi/8$ to $\pm\pi/4$ to have a larger give-way section and more often encourage a pass behind the obstacle. The motivation for this is the possibility of a sudden appearance of a vessel from behind an island. It could be more appropriate to trigger a behind pass in this case than had it been noted earlier. As the modification of this angle simultaneously increases the angle for stand-on, another option would be to choose the angle asymmetrical.

If a non-safe situation is detected, the situation state is kept as this until the corresponding exit criterion is fulfilled, or the emergency criterion is met. The exit criteria are implemented as follows:

- HO, GW, and EM are considered resolved once the ego vessel is located (central) aft of the obstacle vessel and it is no longer approaching:

$$(t_{\mathrm{CPA}} < 0) \wedge (\phi_{\mathrm{BA}} \in [5\pi/8, 11\pi/8]) \tag{13}$$

- SO is considered resolved once the obstacle vessel is located (central) aft of ego vessel and it is no longer approaching:

$$(t_{\mathrm{CPA}} < 0) \wedge (\phi_{\mathrm{AB}} \in [5\pi/8, 11\pi/8]) \tag{14}$$

- OT is considered resolved once the vessel is no longer approaching and the current distance between the vessels, d_{AB}, is greater than d_{SF}:

$$(t_{\mathrm{CPA}} < 0) \wedge (d_{\mathrm{AB}} < d_{\mathrm{SF}}) \tag{15}$$

Emergency mode is entered once the collision risk is critical, defined as

$$(d_{\mathrm{CPA}} < d_{\mathrm{crit}}) \vee (0 < t_{\mathrm{CPA}} < t_{\mathrm{crit}}) \tag{16}$$

where d_{crit} and t_{crit} are parameters. This definition is similar to the entry criteria in [11], [12]. As the planner only considers obstacles in front of the ego vessel, any vessel approaching from aft would here not trigger an evasive maneuver.

Although such a mode is not further described here, once no vessels are in a non-safe state, a return to a pure path-following mode could be considered.

D. Defining Obstacles

Dynamic obstacles, and smaller static obstacles not considered by $\mathbf{X}_\mathrm{f}$, are added to the optimization problem as elliptical constraints described in Eq. (5). For the dynamic obstacles, the shape of the representation is decided by the identified situation and the size is set using the radius r_SF. This radius may vary depending on the available space, as further described in Sec. III-F. Smaller static obstacles extracted from the map are preferably classified as static upon creation, in order not to be eligible for the COLREGs situation identification.

In Fig. 5, example ellipses are shown for the different situations. Based on r_SF the ellipse radii for short and long edge, r_s and r_l, respectively, are described by

$$r_\mathrm{s} = 1.5 r_\mathrm{SF}, \quad r_\mathrm{l} = 3 r_\mathrm{s}, \quad r_\mathrm{off} = \frac{2}{3} r_\mathrm{l} \tag{17}$$

For GW, the long edge is in the direction of travel, r_x, to avoid passing ahead of the obstacle vessel, and for HO it is in the sideways direction, r_y, to encourage a pass on the port side of the obstacle vessel. For both, the ellipse center is r_off away from the obstacle position. The states SO and SF have no restricted area defined.

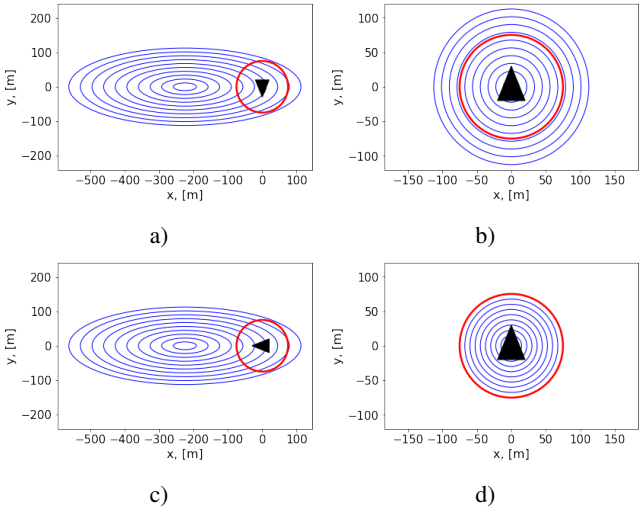

Fig. 5. Based on the radius r_SF = 75 m (marked with the red circle), elliptic restricted areas (blue) are created depending on the situation. Here the areas are illustrated with level curves showing the cost increase for intrusion based on σ_obs. The location and direction of the obstacle is marked by the black triangle (larger than the expected vessel size for illustrative purposes). a) Head-on, HO b) Overtake, OT c) Give-way, GW d) Emergency, EM.

The EM entry criterion is defined in Eq. (16). With r_min defining the lower limit of r_SF, let $d_\mathrm{crit} \leq r_\mathrm{min}$ so that a successful avoidance, given the restricted areas for HO, GW, or OT, will not trigger EM. The restricted area for EM is a circle with radius r_SF with the obstacle located at the center. This opens up for a tighter passing on any side of the obstacle, and centers the maximum cost to the middle of the restricted area where the obstacle is located.

E. Stand-On Scenario

If any obstacle is causing the ego vessel to stand on, the nominal path is temporarily replaced. The direction of the new path is set by the current course of the ego vessel and starts at the current position at the time of SO entry. The temporary path itself may ignore the static map, and extend over islands, since the pursuit point itself will, per definition, be located within $\mathbf{X}_\mathrm{f}$. The obstacle causing SO will not be accompanied by a restricted area unless EM is detected. In case of multi-vessel encounter, all other vessels may cause evasive maneuvers from the temporary stand-on path. The free-space identification scheme (Sec. III-B) will, in step 2.2, use parallel expansion along both the nominal path and the stand-on path. This is to ensure an efficient return to the nominal plan, once the situation is resolved, see Fig. 6. The box expansion, in step 2.3, is centered around the stand-on path.

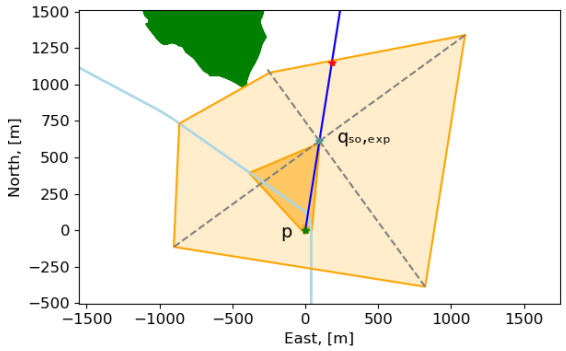

Fig. 6. In a stand-on scenario, a stand-on path, in blue, is assigned in addition to the nominal path, in light blue. This affects the resulting free space (outer orange) and the pursuit point (red).

F. Scalability in Obstacles

The distance required to stay clear of other vessels is not explicitly stated in the COLREGs. The concept of good seamanship is always to be considered. Here, this is interpreted as that a minimum safe distance should always be maintained, but a larger distance is desirable when space permits. The suitable distance to keep clear is therefore adjusted depending on the environment. The radius $r_\mathrm{SF} \in [r_\mathrm{min}, r_\mathrm{max}]$, where r_min and r_max are user parameters, is scaled with respect to the surrounding environment. By creating a bounding box around $\mathbf{X}_\mathrm{f}$ and taking the smaller width b_min as a measure, together with a scaling factor, α, and l_max, a scaled candidate, $\hat{r}_\mathrm{SF}$, for r_SF is computed as

$$\hat{r}_\mathrm{SF} = \max(r_\mathrm{min}, r_\mathrm{max} \min(1, \alpha b_\mathrm{min}/l_\mathrm{max})) \tag{18}$$

There is a memory in r_SF according to

$$r_\mathrm{SF} = \min(\hat{r}_\mathrm{SF}, r_\mathrm{SF,old}) \tag{19}$$

where $r_\mathrm{SF,old}$ denotes the previous value of r_SF, such that over time the radius cannot increase, only decrease. A sudden increase could potentially have the ego vessel to end up further inside the created restricted area and cause unwanted discontinuities in the plan.

IV. Simulation and Experiments

The developed method has been tested and evaluated both in simulations and in field experiments[1]. For all, the optimization problem Eq. (3) has been solved using the solver Ipopt [17] in CasADi [18]. The results presented here are in selected scenarios for demonstration of the method. There are numerous scenarios that can be created, and methods to evaluate the overall performance of such an algorithm and this is research of its own as discussed in, *e.g.*, [4] and [5]. In order to execute the plan, the initial course and speed provided by the planner is, for simplicity, used as set points for course and speed until the next planning iteration. In the simulations, the set points are implemented as the actual course and speed and no boat dynamics are considered.

The parameters used for simulations and field experiments are listed in Table I. The horizon H_p was chosen heuristically based on performance in computation and such that $(2l_{\max}/H_\mathrm{p}) < r_{\min}$ in order to keep the step size dense enough for the model predictions.

A. Simulations

In the first presented simulation, a two-vessel crossing situation is considered. The blue vessel B2 in Fig. 7 is to give way to the red vessel B1, which is to stand on. It can be seen that the give-way plan for B2, illustrated with blue dots, uses the available space, marked with orange, but in addition, as maneuver space is limited, it will need to slow down to let the obstacle pass. The grey ellipse illustrates the restricted area around the obstacle vessel B1. The plan for B1 is a stand-on plan and does not follow the red vessel's original plan but aims straight ahead. The closest distance between the vessels in this scenario was 86.0 m and $r_{\min}$ was set to 75 m.

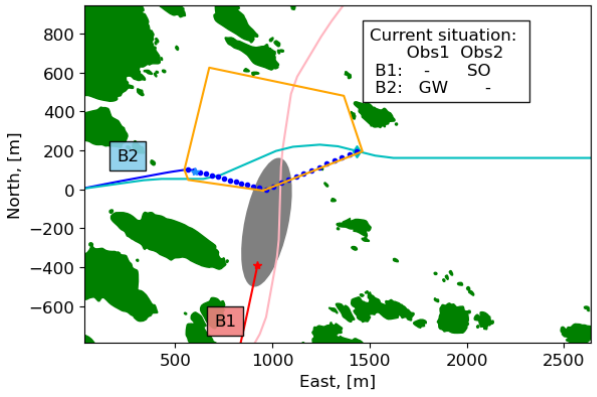

Fig. 7. Simulated scenario with give-way situation for B2. The identified free space is not sufficient for an evasive maneuver without slowing down. The vessel speed will decrease from 4.2 m/s to 3.0 m/s, until the obstacle has passed. The speed decrease corresponds to higher density of way-points in the first part of the plan.

The second presented simulation scenario involves three vessels, all implemented to use the proposed planning method. Two vessels, B2 and B3, are here in a head-on situation and B1 creates a crossing situation with each of the other two as presented in Fig. 8. Vessel B1 is to give way to B2 and stand on in relation to B3. B2 is in a head-on situation with B3 and in a stand-on situation with B1. B1 first detects the give-way situation with B2 and then, once the evasive maneuver is initiated, it detects the stand-on situation with B3. This causes B1 to continue in that direction until the stand-on situation is over. Would B1 have had detected the stand-on situation with B3 prior to starting to give way to B2, a stand-on path would be set before starting to give way. In this scenario, neither of the vessels had to slow down, and the resulting minimum distance was 87.8 m, which is higher than $r_{\min} = 75$ m. The average computation time per vessel at each planning instance was 2.3 s, which is considered reasonable for real-time experiments.

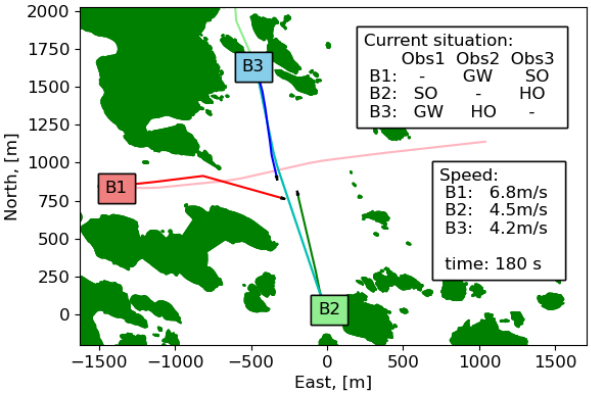

Fig. 8. Scenario with three vessels at time $t = 180$ s. Each vessel B1–B3 considers the other two as obstacles.

B. Field Experiments

The field experiments were performed in the WASP Research Arena for Public Safety, WARA-PS, [19], located at Gränsö, Västervik on the Swedish east coast. The real-time implementation of the planner utilized the Robot Operating System, ROS[2]. The first experiment was a head-on scenario that utilized a Piraya USV [19] together with a simulated, non-acting, obstacle traveling straight as presented in Fig. 9. A down-scaling of the distance parameters was done to accommodate the available space and safety distances. The planning interval T_I was lowered to 5 s in order to enable a quicker response to the planned turning action and still accommodate the estimated planning computation time. The experiment aimed at showing feasibility in a real-time scenario.

In the second experiment, using WARA-PS MiniUSVs[3], both vessels utilized the planner. The scenario was here scaled down, which amplified the impact of environmental disturbances and boat dynamics significantly (Fig. 10). Given that the detection range was set very low, there was not

[1]https://youtu.be/z6MT5hYA4AE [Accessed 2024-04-03]

[2]https://www.ros.org [Accessed 2024-03-07]

[3]https://portal.waraps.org/page/mini-usv [Accessed 2024-03-07]

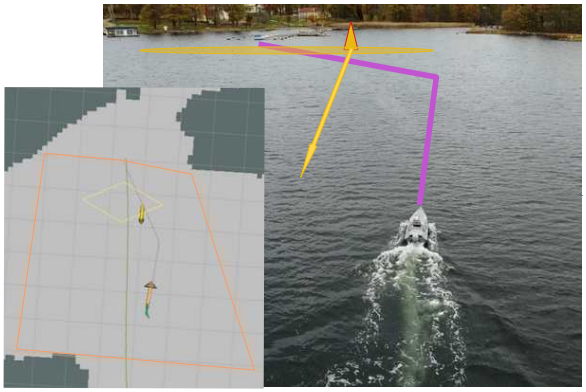

Fig. 9. Field experiment with a head-on scenario. To the left is the scene visualized in RViz, a ROS visualization tool. The grid size is $50\ \mathrm{m} \times 50\ \mathrm{m}$. The yellow romb/ellipse marks the restricted area of the simulated obstacle and the purple line marks the evasive plan.

TABLE I

PARAMETERS FOR SIMULATIONS AND EXPERIMENTS

Parameter	Value[1]	Description
H_p	25	prediction horizon, steps
T_I	25 / 5 / 5 s	planning interval
l_max	1200 / 300 / 50 m	max. look-ahead distance
r_min	75 / 20 / 10 m	min. safety dist.
r_max	300 / 20 / 1was0 m	max. safety distance
d_SF	350 / 350 / 40 m	safe level for d_CPA
d_crit	75 / 20 / 10 m	crit. level for d_CPA
t_SF	300 / 300 / 50 s	safe level for t_CPA
t_crit	30 / 10 / 10 s	crit. level for t_CPA
γ_{v1}	6	weight for J_{v1}
γ_{v2}	10^{-4}	weight for J_{v2}
γ_obs	1000	weight for J_obs
γ_pos	10	weight for J_pos
ε	0.005	par. used in Eq. (5) and (10)
α	0.5	scaling par. used in Eq. (18)

[1]Simulation / Field 1 / Field 2 (if modified)

enough space for an evasive maneuver and, correctly, EM was entered. However, due to the proportionally large impact of disturbances, it can be challenging to maintain a safe distance in this setting.

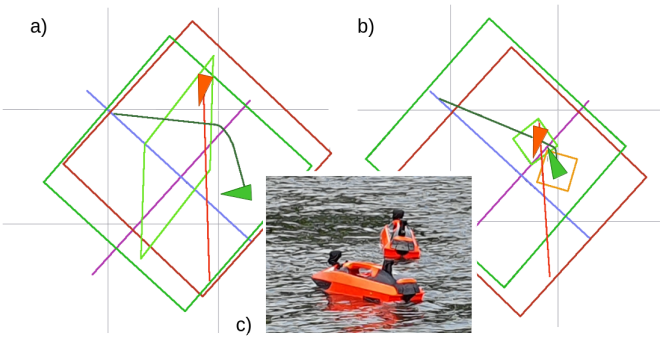

Fig. 10. Field experiment with a two-vessel give-way/stand-on scenario. The green vessel is going north-east and the red vessel is going south-west. The grid size is $50\ \mathrm{m} \times 50\ \mathrm{m}$. a) The red vessel has created a stand-on plan and the green vessel is planning to give way. b) Given the down-scaled problem, the boat dynamics prevents the green vessel from executing the intended maneuver in time and an emergency situation arises. c) Mini-USVs.

V. DISCUSSION

This research aimed to investigate and to propose a local planner for collision avoidance in an archipelago environment with varying available space. The problem statement in Sec. II posed four questions, where the first regarded the irregularly-shaped environment. Representation of irregularly-shaped obstacles in a numerical optimization problem is non-trivial, as argued in [11], where the proposed avoidance method was evaluated in a simplistic world with ellipses representing islands. It is in this research, however, assumed that obstacle vessels are only known in the visual area. This assumption makes it reasonable to only perform evasive planning in the local area using a forward-looking convex approximation of the free space, as defined in Sec. III-B and also illustrated in Fig. 7. The handling of the free space in the proposed planner may be improved so that near-edge situations are prevented for navigational safety and numerical efficiency. Related to this, Bergman *et al.* introduced a penalty distance to static obstacles in the lattice-based planning step [12] and Eriksen *et al.* used a two-level padding of static obstacles [11].

The second item in the problem statement regards adaptation of the problem to the available space. This is primarily solved by the convex approximation of the free space, and having the plan adapt accordingly in time and distance. Should the space be limited, a slow-down maneuver is encouraged as presented in Fig. 7. By introducing a minimum time limit, a slow-down maneuver is also encouraged if the vessel is too close to the pursuit point. Also, a means to adjust clearance distances to other vessels depending on the varying space is introduced in Sec. III-F. Speed is, however, also a factor in safety distances as argued and introduced in [16].

The third problem stated regards the incorporation of COLREGs into the collision-avoidance planner. Given that the situation is detected as GW, HO, OT, or EM these are resolved using spatial constraints in the optimization problem (Sec. III-D) as also done similarly in [11] and [12]. Challenges arise in the situation identification itself, especially in an archipelago setting when vessels may not keep a steady course. Uncertainties in predicted travel are to be expected, but in combination with the non-convex setup of the optimization problem, this may cause discontinuities from one planning iteration to the next (Fig. 10). The decision point on when or not to enter a situational state (including EM) may also cause varying behavior (as in desired maneuver), as the relative bearings vary with the direction of travel. This is further discussed in [6]. The assumption made here to ignore vessels aft of the ego vessel is a simplification as, say, vessels being overtaken should, preferably, also consider to prioritize predictability over path-following. The current limitation in incorporating situational state changes into the predictions used by the numerical optimizer is a challenge, which is also mentioned by Eriksen *et al.* [11]. This means that a currently safe situation may enter a non-safe state, *e.g.*, the

planner may suggest a turn in front of another vessel, without being accounted for in the optimization problem. Bergman *et al.* have, however, incorporated state changes in the lattice planner, which utilizes a search tree rather than numerical optimization [12].

For non-straight routes, as are expected in an archipelago, there is a difference between stand-on and a non-action that motivated the fourth problem stated. Eriksen *et al.* proposed restrictions on the control inputs such that, if any evasive maneuver is needed, it should be readily apparent [11]. This was demonstrated in simulations. Here, the stand-on management is performed via a temporary replacement of the nominal path. A map-based reference, rather than an ego-centered reference, may pose a more robust solution given potential external disturbances such as wind and sea current. Figure 6 provides an illustration on how the design of the free-space polygon allows the vessel to adhere to the guidance of a nominal path even in a stand-on situation. The method was applied in simulations (Figs. 7–8), and a field experiment (Fig. 10).

For the proposed planner in this research, simulations were done to demonstrate the intended behavior. The first simulation, as presented in Fig. 7, shows that the planner has the ability to create valid evasive maneuvers that consider the limited space as well as mission objectives (by ensuring a return back to the nominal path). The planner extends to multi-vessel encounters as presented in the second simulation (Fig. 8). The performed field experiments utilized two different vessel types and demonstrated real-time performance. The down-scaled setting, presented in Fig. 10, demonstrated the method in the presence of deviations between actual and predicted behavior and entering of emergency mode. The emergency mode here only causes a geometrical adaptation, but would additionally benefit from speed reduction.

Further research would include robustness considerations such as introducing fallback solutions, and elaborating on situation transitions and predictions in a curvy-route setting.

VI. Conclusions

In this paper a local collision-avoidance planner for use in archipelago environments was presented. The available space is approximated with a path-centered convex space. The planner is able to adapt evasive maneuvers as well as stand-on behavior according to COLREGs to the available, often limited, space. The planner was implemented and evaluated in both simulations and real-time field experiments.

Acknowledgments

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

The field experiments were supported by the WASP Research Arena for Public Safety, WARA-PS, with skillful technical assistance from Jonas Rosquist, Mattias Nilsson, and Michael Petterstedt.

References

[1] "Convention on the international regulations for preventing collisions at sea, 1972 (COLREGs)," International Maritime Organization, 1972, accessed 2023-02-01. [Online]. Available: https://www.imo.org/en/About/Conventions/Pages/COLREG.aspx

[2] A. Cockcroft and J. Lameijer, *Guide to the Collision Avoidance Rules*. Elsevier Science, Oxford, UK, 2012.

[3] H.-C. Burmeister and M. Constapel, "Autonomous collision avoidance at sea: A survey," *Frontiers in Robotics and AI*, vol. 8, 2021.

[4] P. G. Stankiewicz and G. E. Mullins, "Improving evaluation methodology for autonomous surface vessel colregs compliance," in *OCEANS, June 17-20, Marseille*, 2019.

[5] A. Vagale, "Evaluation simulator platform for extended collision risk of autonomous surface vehicles," *Journal of Marine Science and Engineering*, vol. 10, no. 5:705, 2022.

[6] R. Weber and L. Sánchez-Heres, "COLREG 2 - potential consequences of varying algorithms in traffic situations," Lighthouse, Swedish Maritime Competence Centre, 02 2023, accessed 2023-08-24. [Online]. Available: https://www.researchgate.net/publication/369947822_COLREG_2_-Potential_consequences_of_varying_algorithms_in_traffic_situations

[7] X.-Y. Zhou, J.-J. Huang, F.-W. Wang, Z.-L. Wu, and Z.-J. Liu, "A study of the application barriers to the use of autonomous ships posed by the good seamanship requirement of COLREGs," *The Journal of Navigation*, vol. 73, no. 3, pp. 710–725, 2020.

[8] S. N. MacKinnon, R. Weber, F. Olindersson, and M. Lundh, "Artificial intelligence in maritime navigation: A human factors perspective," in *Advances in Human Aspects of Transportation: Proc. AHFE 2020 Virtual Conf. on Human Aspects of Transportation, July 16-20, 2020, USA*. Springer, 2020, pp. 429–435.

[9] I. B. Hagen, D. K. M. Kufoalor, E. F. Brekke, and T. A. Johansen, "MPC-based collision avoidance strategy for existing marine vessel guidance systems," in *IEEE Int. Conf. Robotics and Automation (ICRA), May 21-26, Brisbane*, 2018, pp. 7618–7623.

[10] P. Stankiewicz and M. Kobilarov, "A primitive-based approach to good seamanship path planning for autonomous surface vessels," in *IEEE Int. Conf. Robotics and Automation (ICRA), May 30-June 5, Xi'an*, 2021, pp. 7767–7773.

[11] B.-O. H. Eriksen, G. Bitar, M. Breivik, and A. M. Lekkas, "Hybrid collision avoidance for ASVs compliant with COLREGs rules 8 and 13–17," *Frontiers in Robotics and AI*, vol. 7, 2020.

[12] K. Bergman, O. Ljungqvist, J. Linder, and D. Axehill, "A COLREGs-compliant motion planner for autonomous maneuvering of marine vessels in complex environments," *arXiv preprint arXiv:2012.12145*, 2021.

[13] C. Rösmann, F. Hoffmann, and T. Bertram, "Timed-elastic-bands for time-optimal point-to-point nonlinear model predictive control," in *European Control Conf. (ECC), July 15-17, Linz*, 2015, pp. 3352–3357.

[14] K. Bergman, O. Ljungqvist, J. Linder, and D. Axehill, "An optimization-based motion planner for autonomous maneuvering of marine vessels in complex environments," in *59th IEEE Conf. Decision and Control (CDC), December 14-18, Jeju Island*, 2020, pp. 5283–5290.

[15] A. B. Martinsen, G. Bitar, A. M. Lekkas, and S. Gros, "Optimization-based automatic docking and berthing of ASVs using exteroceptive sensors: Theory and experiments," *IEEE Access*, vol. 8, pp. 204 974–204 986, 2020.

[16] C. Tam and R. Bucknall, "Collision risk assessment for ships," *Journal of Marine Science and Technology*, vol. 15, pp. 257–270, 2010.

[17] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, pp. 25–57, 2006.

[18] J. A. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi: A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, pp. 1–36, 2019.

[19] O. Andersson, P. Doherty, M. Lager, J.-O. Lindh, L. Persson, E. A. Topp, J. Tordenlid, and B. Wahlberg, "WARA-PS: A research arena for public safety demonstrations and autonomous collaborative rescue robotics experimentation," *Autonomous Intelligent Systems*, vol. 1, no. 9, pp. 1–31, 2021.